

## **BAB II**

### **LANDASAN TEORI**

#### **2.1. Stroop Task**

*Stroop task* merupakan salah satu tes psikologi yang berguna untuk menguji kemampuan *kognitif* dan pengambilan keputusan. *Stroop task* yang paling dasar adalah sebuah *task* yang menguji kemampuan seseorang dalam menghadapi *interferensi* didalam otak. Variabel yang diuji dalam *Stroop task* adalah akurasi dan kecepatan subjek dalam menghadapi *task* yang diberikan. Pada *Stroop task*, *interferensi* terjadi antara tulisan yang akan dibaca, dengan warna tulisan yang dilihat oleh subjek [2].

*Stroop task* dilakukan dengan menunjukan daftar kata-kata nama warna terhadap subjek. Hanya saja tiap kata warna yang diberikan kepada subjek akan memiliki warna tulisan yang berbeda dengan kata yang tertera. Misalkan kata “Biru” akan ditunjukan kepada subjek, tetapi tulisan “Ungu” akan berwarna biru. Subjek akan diminta untuk menyebutkan warna dari tulisan “Ungu” . *Stroop task* mungkin terlihat mudah tetapi proses *interferensi* akan membuat kesulitan bagi masing masing subjek karna masing-masing tulisan tidak *kongruen* [7].[8].

Tugas *Stroop* adalah mengaktifkan dua jalan pada satu waktu, yaitu kemampuan subjek mendeteksi warna yang dilihat, dan kemampuan subjek mendeteksi tulisan yang dibaca. Gangguan akan terjadi jika dua hal tadi aktif secara bersamaan sehingga terjadi *interferensi* satu sama lain [8].

#### **2.2. Selective Attention Dan Memori.**

*Selective attention* atau perhatian yang selektif adalah salah satu dari pembagian *atensi* (perhatian), seseorang akan diberikan instruksi untuk memberikan perhatian yang sama pada tugas. Pada selektif perhatian/*atensi*, seseorang akan dilawankan dengan dua atau lebih tugas yang serempak dan dibutuhkan untuk memfokuskan perhatian mereka pada salah satu serta mengabaikan yang lain atau secara sederhana *selective attention* adalah bagaimana kita memusatkan perhatian pada objek untuk meningkatkan daya konsentrasi yang kita miliki [8].

Memori atau disebut juga ingatan adalah suatu daya yang dapat menerima, menyimpan, dan memproduksi kembali informasi yang sudah lampau. Definisi dari Schlessinger dan Groves (1976) adalah suatu sistem yang sangat berstruktur, yang menyebabkan organisme sanggup merekam fakta tentang dunia dan menggunakan pengetahuannya untuk membimbing perilakunya. Setiap saat *stimulasi* mengenai indra kita, setiap saat pula *stimulasi* itu direkam secara sadar atau tidak sadar. kemampuan rata rata memori manusia bahkan mencapai satu *kuidrilliun* bit informasi [8].

Secara sederhana hubungan antara perhatian yang *selektif* dan memori adalah Pada saat seseorang ingin meningkatkan fungsi memorinya maka seseorang juga harus memusatkan perhatian atau fokus pada sesuatu objek yang ingin diingatnya . semakin banyak seseorang melatih daya ingatnya maka akan semakin baik pula daya simpan memori yang dia miliki dan akan semakin tinggi pula *selective attention* dan konsentrasi pada dirinya. Pada saat seseorang berkonsentrasi dan memperhatikan sesuatu maka otak akan terbagi menjadi dua sistem yaitu *reaktif* dan terfokus, *reaktif* merupakan sistem pusat emosi yang pasif dan tanpa sadar sedangkan terfokus merupakan sistem yang berpusat pada *eksekutif* otak, *korteks* depan yang bertanggung jawab melakukan perencanaan, pengaturan, dan antisipasi konsekuensi tindakan yang diambil [8].

### **2.3. Brain Computer Interface**

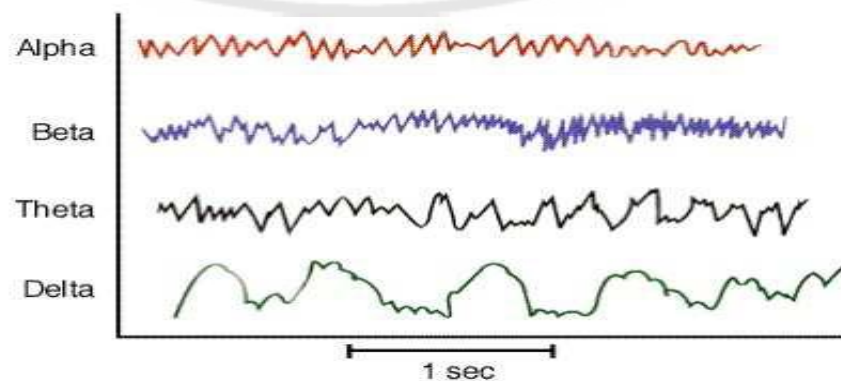
*Brain-Computer Interface* merupakan suatu ilmu yang mempelajari kemungkinan otak untuk dapat berinteraksi langsung dengan perangkat keras, dengan bantuan alat penerima sensor gelombang otak. Gelombang otak yang didapat biasa disebut dengan *electrical biosignal*. *Electrical biosignal* merupakan arus listrik yang dihasilkan oleh perbedaan potensial listrik di sistem jaringan, organ atau sel seperti sistem saraf. Contoh dari *electrical biosignal* adalah *ECG* (*electrocardiogram*), *EMG* (*electromyogram*), *EEG* (*electroencephalogram*) dan *EOG* (*electrooculogram*). Dari *electrical biosignal* diatas yang umum digunakan adalah *electroencephalogram* (*EEG*) yang biasanya diukur dengan cara menempelkan *eletroda* ke kulit kepala. [9]

*Brain Computer Interface* adalah perangkat *eksternal* yang dapat digunakan langsung untuk berkomunikasi dengan *brain/otak* manusia atau hewan melalui antar muka khusus (*silikon neuron*). Signal gelombang yang dihasilkan oleh otak di rekam oleh komponen *elektrik* yang disebut *electroda* yang ditempelkan pada posisi tertentu di kulit kepala terhubung ke perangkat *electroencephalograph* (*EEG*) dapat diidentifikasi sebagai suatu instruksi yang harus dilakukan oleh organ-organ.[9]

*Electroencephalogram* (*EEG*) adalah rekaman aktivitas listrik sepanjang kulit kepala yang dihasilkan oleh penembakan neuron dalam otak. *EEG* mengacu pada rekaman aktivitas spontan *elektrik* otak dalam waktu singkat biasanya 20 - 40 menit yang direkam dari beberapa *elektroda* ditempatkan pada kulit kepala.[9]

*Brain Computer Interface* didefinisikan sebagai sebuah sistem yang terdiri dari dua *entitas* yaitu subjek orang dan subjek mesin. Setiap subjek melakukan komunikasi melalui perangkat *EEG* kemudian mesin melakukan respon dengan suatu tindakan atau gerakan [10]. Proses komunikasi subjek dengan cara memproduksi sinyal *EEG* dan mesin melakukan respon untuk mengerjakan pekerjaan. Sinyal yang dihasilkan oleh gelombang otak manusia dengan sebutan sinyal *EEG* dihasilkan oleh komputer sebagai proses sinyal dan aksi yang ditimbulkan divisualisasikan dalam bentuk sinyal multimedia dinamis (objek 3D, gambar, video, dan suara) [10].

*EEG* biasanya diilustrasikan sebagai pita frekuensi. Pita frekuensi pada *EEG* terbagi menjadi beberapa frekuensi, yaitu *alpha*, *beta*, *gamma*, *delta*, dan *theta*. dan dapat dilihat dalam Gambar 2.1



Gambar 2.1 Jenis-jenis frekuensi *EEG*

*EEG* bekerja dengan menangkap frekuensi sinyal-sinyal listrik yang dibangkitkan oleh otak akibat adanya aktivitas mental subjek. Dalam analisis untuk tujuan klasifikasi sinyal *EEG*. Pada umumnya digambarkan dalam pita frekuensi. Amplitudo *EEG* menunjukkan banyak variasi pada rangsangan eksternal dan juga kondisi mental internal. *Delta, Theta, Alpha, Beta, dan Gamma* merupakan nama-nama dari *brainwave* atau pita frekuensi *EEG* yang berkaitan dengan variasi kondisi otak yang diklasifikasikan dalam Tabel 2.1

Tabel 2.1 Klasifikasi Kondisi Mental

<b>Tipe Gelombang Otak</b>	<b>Range Frekuensi</b>	<b>Kondisi Mental</b>
<b>Delta</b>	0.5 Hz – 4 Hz	Bawah sadar, tidur tanpa mimpi, tidur non-REM ( <i>Rapid Eye Movement</i> )
<b>Theta</b>	4 Hz -7 Hz	Mimpi, berfantasi, mengingat-ingat
<b>Alpha</b>	8 Hz – 12 Hz	Sadar, rileks tapi tidak mengantuk, tenang
<b>Low Beta</b>	12 Hz – 15 Hz	Rileks tapi fokus, SMR ( <i>Sensory Motor Rhythm</i> )
<b>Midrange Beta</b>	16 Hz – 20 Hz	Berpikir, sadar akan kendali diri dan sekitar
<b>High Beta</b>	21 Hz – 30 Hz	Waspada, gelisah
<b>Gamma</b>	30 Hz – 100+ Hz	Fungsi motorik, aktivitas mental yang lebih tinggi

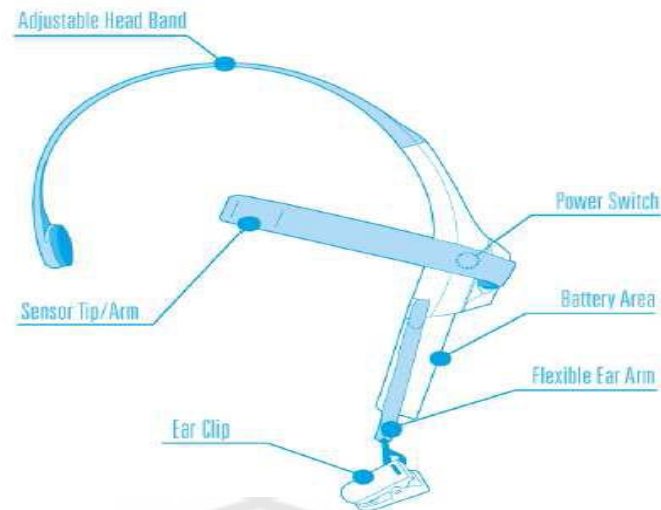
#### 2.4. Neurosky Mindwave

*Neurosky* merupakan perusahaan yang berdiri sejak 2004 di Silicon Valley, California, Amerika Serikat. Perusahaan ini berfokus pada tujuan utamanya yaitu memanfaatkan teknologi *Brain Computing interface (BCI)* untuk dipasarkan pada konsumen secara luas. *Neurosky* mengadaptasi teknologi *EEG* dan mengembangkannya untuk dapat digunakan dalam beberapa bidang yang memenuhi permintaan komersial, seperti untuk *entertainment*, kesehatan, serta otomotif [11]. Terobosan terbesar *Neurosky* adalah ketika mereka berhasil

mengembangkan perangkat *EEG* dengan harga yang cukup murah. Perangkat tersebut meliputi : sensor kening yang berfungsi menangkap input sinyal dan gelombang yang dihasilkan oleh otak, perangkat lunak dan perangkat keras yang mampu mengurangi dan memfilter setiap *elektrikal noise*, serta *think gear chip* yang berfungsi sebagai sirkuit pemrosesan sinyal dan penghasil *output*. Seluruh perangkat tersebut dapat ditanamkan pada perangkat sederhana serupa *headset* [11].

Sinyal *attention* mengindikasikan intensitas tingkat fokus pengguna seperti yang terjadi selama konsentrasi dan diarahkan. Mempunyai *range* dari 0 hingga 100. Gangguan sekitar, melamun, kurang konsentrasi, atau cemas dapat menurunkan tingkat *attention*. Sedangkan sinyal *meditation* mengindikasikan level mental relaksasi pengguna yang mempunyai *range* 0 hingga 100 juga. Perlu diketahui bahwa sinyal *meditation* adalah ukuran dari kondisi mental seseorang, bukan tingkat fisik, sehingga hanya dengan meregangkan otot tubuh saja tidak akan langsung menghasilkan tingkat *meditation* tinggi. Namun bagi kebanyakan orang, merilekskan tubuh akan membantu pikiran untuk rileks juga. *meditation* berhubungan dengan penurunan aktivitas mental pada otak. Telah dilakukan pengamatan bahwa dengan menutup salah satu mata akan mematikan aktifitas mental proses gambaran oleh mata. Jadi dengan menutup mata akan menjadi metode efektif untuk meningkatkan tingkat *meditation* [11].

*Mindwave* mengukur tegangan antara sensor yang diletakkan di dahi dan sensor yang menjepit telinga kiri (*ear clip*) sebagai ground. Lebih tepatnya posisi pada dahi adalah Fp1, sebagaimana yang telah ditetapkan oleh sistem internasional 10-20 [11]. Selanjutnya, tidak seperti *EEG* lain yang berada di laboratorium medis, *Mindwave* tidak memerlukan gel atau sejenisnya untuk perekaman, dan tidak membutuhkan keahlian khusus untuk mengoperasikannya. Selain itu alat ini sudah memiliki *Noise Cancellation Technologies* yang dapat menghilangkan frekuensi *noise* dari sumber-sumber lain seperti gerakan otot dan perangkat listrik. Filter tersebut menghilangkan gangguan listrik yang bervariasi dari 50 Hz hingga 60 Hz tergantung pada lokasi geografis. Berikut adalah bagian-bagian dari alat *Neurosky Mindwave* seperti yang terlihat dalam Gambar 2.2.

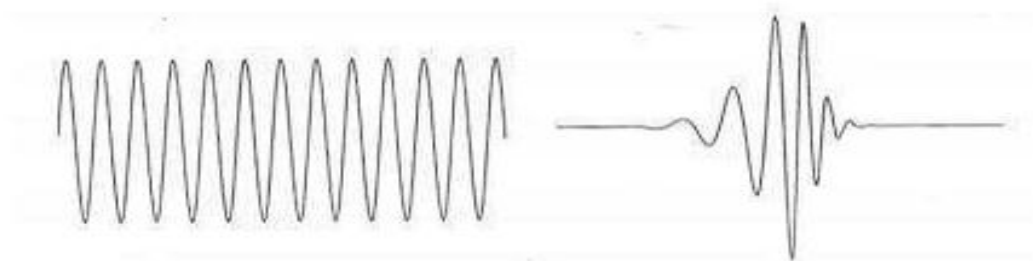


Gambar 2.2 Bagian-bagian dalam *Neurosky Mindwave*

*Neurosky Mindwave Headset* ini terdiri beberapa bagian penting yang dapat mempengaruhi hasil pembacaan yaitu klip pada telinga (*ear clip*), sensor yang berada pada ujung lengan (*sensor arm*) dan tiga bagian yang berada di ujung (*sensor arm*), dan lengan sensor. Referensi dan *ground* elektroda *headset* berada pada klip telinga (*ear clip*) dan sensor elektroda yang digunakan untuk membaca gelombang otak berada pada pengait depan (*sensor arm*) yang bertumpu pada dahi di atas mata [11].

## 2.5. Wavelet

*Wavelet* adalah sebuah gelombang kecil, yang dimana energinya terkonsentrasi dalam waktu untuk menyediakan alat bantu analisis *non-stationer* atau perubahan waktu. Karakteristik *wave* bergerak masih tetap dimiliki, namun juga dapat mensimulasikan analisis waktu-frekuensi dengan dasar matematika yang fleksibel. Bentuk sebuah gelombang dan *wavelet* ditunjukkan pada Gambar 2.3



Gambar 2.3 Bentuk sebuah *Wave* dan *Wavelet*.

Pada gambar 2.3 dimana *wave* (kurva sinus) bergerak dengan amplitude sama pada  $-\infty \leq t \leq \infty$  sehingga memiliki energi yang tidak berhingga, dengan *wavelet* yang memiliki energi berhingga terkonsentrasi pada suatu titik.[12]

### 2.5.1 *Wavelet Transform*

Sinyal *EEG* yang dihasilkan *Neorosky Mindwave* merupakan jenis sinyal *non-stationer* atau sinyal yang memiliki frekuensi yang bervariasi didalam waktu. Sehingga untuk menganalisanya dibutuhkan metode transformasi yang dapat memberikan resolusi frekuensi dan waktu secara bersamaan. *Wavelet* adalah salah satu metode yang cocok memberikan resolusi frekuensi dan waktu secara bersamaan dikarenakan *transformasi wavelet* dapat mempresentasikan informasi suatu sinyal dalam kawasan waktu dan frekuensi dengan baik [13].[14].

Setiap *wavelet* memiliki karakteristik lokasi dan skala [14]. Basis *wavelet* berasal dari sebuah fungsi penskalaan atau dikatakan juga sebuah *scaling function*. *Scaling function* memiliki sifat yaitu dapat disusun dari sejumlah salinan dirinya yang telah didilasikan, *ditranslasikan* dan diskalakan. Menurut [14], *wavelet* merupakan gelombang mini (*small wave*) yang mempunyai kemampuan mengelompokkan energi citra dan terkonsentrasi pada sekelompok kecil koefisien, sedangkan kelompok koefisien lainnya hanya mengandung sedikit energi yang dapat dihilangkan tanpa mengurangi nilai informasinya.

*Wavelet* dibagi menjadi dua berdasarkan ruang dan waktu yaitu *wavelet 1D* (waktu) dan *2D* (Ruang). Pengertian waktu disini adalah untuk gelombang 1D, pada saat memulai *point shifting* dari sumber menuju akhir, sedangkan pengertian ruang didalam *wavelet 2D*, *point shiftingnya* adalah dua dimensi.

Secara matematis fungsi *wavelet* adalah memisahkan data menjadi beberapa komponen frekuensi yang berbeda dan mempelajari tiap-tiap komponen sesuai dengan resolusi yang tepat dengan skalanya masing-masing. *Wavelet transform* didefinisikan dari fungsi *scaling* dan fungsi *wavelet*. *Wavelet transform* bekerja dengan mengambil rata-rata dari nilai masukan dan mempertahankan informasi yang diperlukan untuk mengembalikan ke nilai semula. Dalam analisis multiresolusi (AMR), *scaling function* digunakan untuk menciptakan satu

rangkaian pendekatan pada suatu fungsi atau citra. Suatu skala yang digunakan dalam *transformasi wavelet* adalah kebalikan dari frekuensi. Skala yang kecil menandakan frekuensi tinggi dan sebaliknya. Pada *transformasi wavelet*, semua fungsi *wavelet* berelasi dengan satu *wavelet* tunggal, yang dikenal dengan ibu *wavelet* [14]

### 2.5.2 Dekomposisi Wavelet

*Wavelet* dapat digunakan untuk melakukan analisis multi resolusi yang akan menghasilkan informasi dalam ranah waktu dan frekuensi. Skala atau resolusi yang biasanya dilihat pada data merupakan peranan yang penting. Algoritma *wavelet* memproses data pada skala atau resolusi yang berbeda-beda. Jika sebuah sinyal dengan jendela yang besar, maka seseorang hanya akan memperhatikan informasi sinyal secara general, begitu juga saat sinyal dengan jendela yang kecil maka seseorang hanya akan memperhatikan sinyal pada detailnya saja, sehingga penggunaan resolusi yang bervariasi sangat diperlukan. Dasar dari prosedur analisis *wavelet* adalah pemilihan fungsi *prototype* yang disebut *mother wavelet*. Analisis sementara dilakukan dengan frekuensi tinggi yang merupakan versi dari *prototype wavelet*, sedangkan untuk analisis frekuensi dilakukan dengan dilatasi pada frekuensi rendah dari *wavelet* yang sama [15].

### 2.5.3 Transformasi Wavelet Kontinyu

Transformasi *wavelet* kontinyu didefinisikan secara matematis pada persamaan 2.1

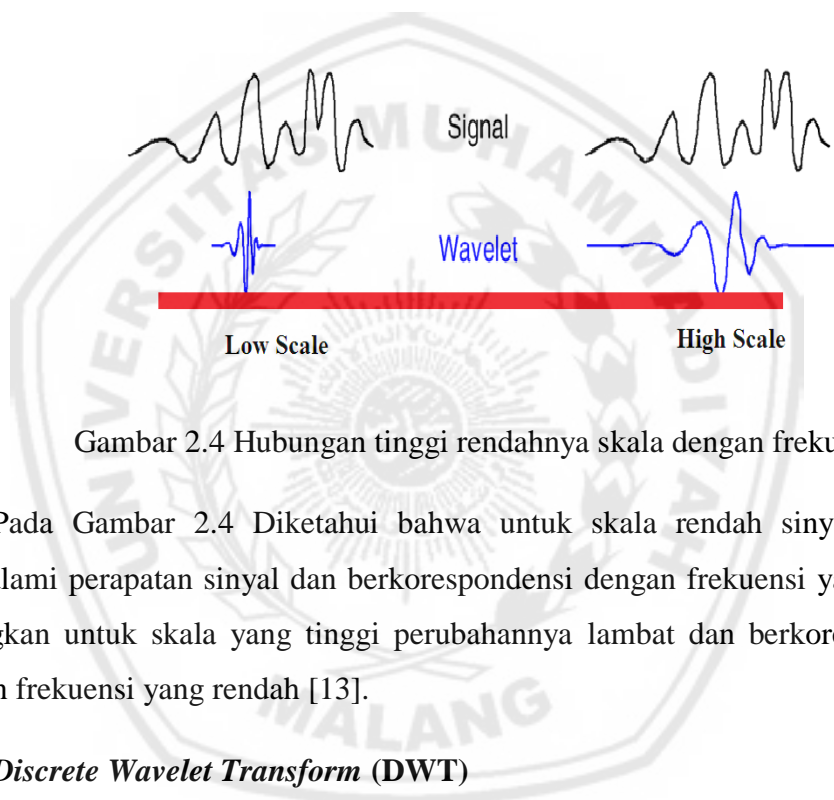
$$y(s, \tau) = \int f(t) \psi_{s,\tau}^*(t) dt \quad 2.1$$

Dimana  $\psi_{s,\tau}^*$  adalah fungsi *wavelet* penganalisa. Persamaan ini menunjukkan bagaimana fungsi  $f(t)$  didekomposisikan kedalam sebuah set dari fungsi basis  $s, \psi(t)$  disebut dengan *wavelet*. Variabel  $s$  dan  $\tau$  merupakan skala dan translasi adalah dimensi baru setelah ditransformasi. *wavelet* diperoleh dari sebuah *wavelet* dasar yang disebut *mother wavelet* [15]



Cara kerja transformasi wavelet kontinyu adalah dengan menghitung *konvolusi* sebuah sinyal dengan sebuah jendela modulasi pada setiap waktu dengan tiap skala yang diinginkan. Jendela modulasi mempunyai skala *fleksibel* dan biasa disebut induk *wavelet* atau fungsi dasar *wavelet* [13].

Transformasi wavelet kontinyu memiliki skala dan frekuensi yang lebih tinggi berkorespondensi dengan *wavelet* yang paling renggang. *wavelet* yang lebih renggang merupakan sinyal kasaran utama yang diukur oleh koefisien *wavelet*. Untuk lebih jelasnya, dapat terlihat dalam Gambar 2.4.

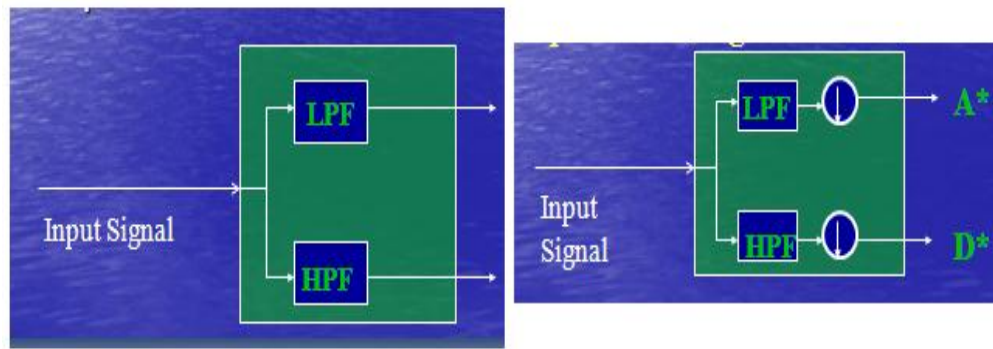


Gambar 2.4 Hubungan tinggi rendahnya skala dengan frekuensi

Pada Gambar 2.4 Diketahui bahwa untuk skala rendah sinyal *wavelet* mengalami perapatan sinyal dan berkorespondensi dengan frekuensi yang tinggi, sedangkan untuk skala yang tinggi perubahannya lambat dan berkorespondensi dengan frekuensi yang rendah [13].

#### 2.5.4 Discrete Wavelet Transform (DWT)

*Discrete Wavelet Transform* merupakan *wavelet* yang merepresentasikan sinyal dalam domain waktu dan frekuensi. *Discrete Wavelet Transform* memiliki keunggulan diantaranya adalah mudah diimplementasikan dan efisien dalam hal waktu dan perhitungan. Pada transformasi ini terjadi *filterisasi* dan *down sampling*, yaitu pengurangan koefisien pada fungsi genap. Hal ini bisa diuraikan dalam Gambar 2.5



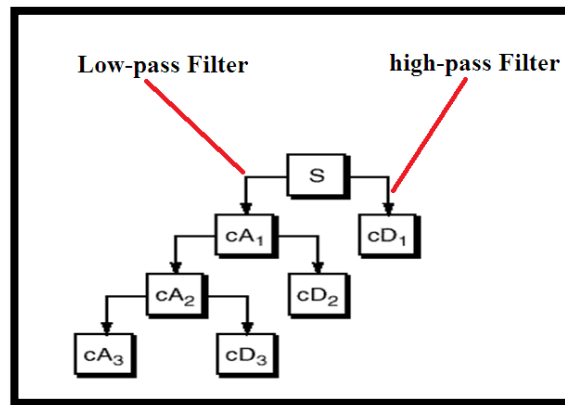
Gambar 2.5 Komponen *LPF*, *HPF*, proses *filterisasi* dan *down sampling*

Prinsip dasar *Discrete Wavelet Transform* adalah bagaimana cara mendapatkan representasi waktu dan skala dari sebuah sinyal menggunakan sebuah teknik *pemfilteran* digital dan operasi *sub-sampling* [14]. untuk menaksirkan siny hasil *transformasi* ini dibentuk *filter low pass* dan *high pass*. Keluaran dari *high pass* dan *low pass* bisa dilihat dalam bentuk Persamaan 2.2 dan 2.3

$$y_{HP}[k] = \sum u[m] \cdot g[2k - m] \quad 2.2$$

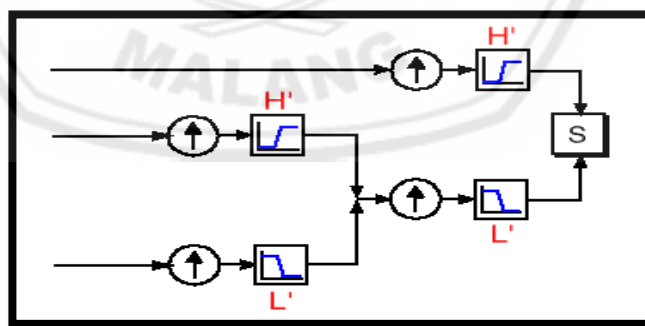
$$y_{LP}[k] = \sum u[m] \cdot h[2k - m] \quad 2.3$$

Pada tahap awal sinyal (S) akan dilewatkan pada rangkaian *high pass filter* dan *low pass filter*, kemudian setengah dari masing-masing keluaran diambil sebagai sampel melalui operasi *sub-sampling*. Proses ini disebut sebagai proses dekomposisi satu tingkat. Keluaran dari *low pass filter* digunakan sebagai masukan dan akan diproses pada dekomposisi berikutnya. Proses ini akan diulang sampai tingkat proses dekomposisi yang diinginkan. Gabungan dari keluaran-keluaran *high pass filter* dan *low pass filter* yang terakhir disebut sebagai koefisien *wavelet* dan berisi informasi sinyal hasil transformasi yang telah terkompresi. Proses dekomposisi secara detail dapat dilihat dalam gambar 2.6

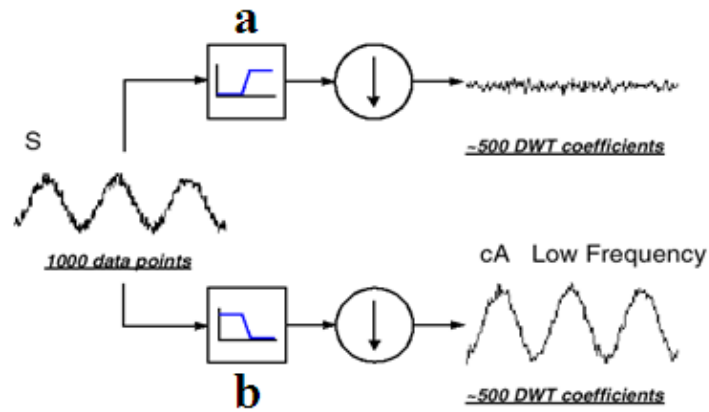


Gambar 2.6 Proses Multi-level decomposition

Didalam *Discrete Wavelet Transform* terdapat proses pengembalian kembali komponen-komponen yang telah kita gunakan. *invers Discrete Wavelet Transform (IDWT)* merupakan kebalikan dari *Discrete Wavelet Transform (DWT)*. Pada transformasi ini dilakukan proses rekonstruksi sinyal atau mengembalikan komponen frekuensi menjadi komponen sinyal semula. Transformasi dilakukan dengan proses *up sampling* dan pemfilteran dengan koefisien filter *invers*. Sehingga dalam satu sistem *transformasi wavelet* empat macam filter, yaitu *low pass filter* dan *high pass filter* dekomposisi, dan *high pass filter* rekonstruksi. Proses rekonstruksi transformasi secara detail digambarkan dalam Gambar 2.7 dan 2.8



Gambar 2.7 Proses rekonstruksi Transformasi Wavelet Diskrit (DWT) dan Invers Diskrit Wavelet Transform (IDWT)



Gambar 2.8 (a) Proses DWT dan (b) Proses IDWT menggunakan data wavelet db2

### 2.5.5 Discrete Wavelet Transform Biorthogonal

Ketika input data ke algoritma terlalu besar untuk diproses, input data tersebut dapat dirubah ke dalam fitur yang telah direduksi. Proses ini disebut *ekstraksi fitur EEG*, *ekstraksi fitur* dilakukan dengan menggunakan transformasi *wavelet*. Tetapi tersedia banyak macam multiple *wavelet* dalam keluarga *wavelet*, selain itu *wavelet* yang sesuai harus dipilih untuk *ekstraksi* yang efisien dari *fitur EEG* yang berbeda. kebanyakan *fitur EEG* terdapat gelombang frekuensi yang berbeda, yaitu gelombang *alpha*, *beta*, *gamma*, *delta*, *theta*.

Transformasi *wavelet* memiliki kelebihan seperti pembatasan frekuensi waktu, penyaringan nilai ganda, dan analisa ruang skala. Transformasi *wavelet* menggunakan sebuah *variable* window pada panjang sinyal. *DWT* sering diperkenalkan dalam istilah *recovery transform*. *Recovery transform* dijelaskan secara detail dalam persamaan 2.4

$$x(t) = \sum_k^{\infty} = -\infty \sum_l^{\infty} = -\infty d(k, l) 2^{-\frac{k}{2}} \psi(2^{-k}t - l) \quad 2.4$$

Disini  $k$  berkaitan dengan  $a$  sebagaimana:  $a = 2^k$ ;  $b$  berkaitan dengan  $l$  sebagaimana  $b = 2^k - 1$ ; dan  $(k, l)$  adalah sample dari  $W(a, b)$  pada titik yang berlainan  $k$  dan  $l$  [14].

*Biorthogonal wavelet* adalah transformasi *wavelet* yang berkaitan dalam *invertible* tetapi belum tentu *orthogonal*, FK dan gk dikatakan *biorthogonal* jika memiliki persamaan seperti persamaan 2.5

$$(f_j, g_k) = \delta_{jk}, \quad 2.5$$

Urutan untuk membangun *wavelet* dapat dilihat dalam persamaan 2.6 dan 2.7

$$\psi_{j,k} = 2^{\frac{j}{2}} \psi(2^j x - k) \quad 2.6$$

$$\widetilde{\psi}_{jk} = 2^{\frac{j}{2}} \psi(2^j x - k) \quad 2.7$$

untuk membangun (2.5) dan (2.6) filter  $g, h, \bar{g}, \bar{h}$ , diperlukan. Dua urutan dekomposisi  $g_n$  dan  $h_n$  berlaku sebagai pembangunan urutan jika  $c_n^1$  adalah seperangkat data, selanjutnya bisa didekomposisikan berdasarkan persamaan 2.8 dan 2.9 [14]

$$c_n^0 = \sum_k g_{2n-k} c_k^1 \quad 2.8$$

$$d_n^0 = \sum_k h_{2n-k} c_k^1 \quad 2.9$$

Untuk rekonstruksi, Syarat untuk rekonstruksi yang sempurna berdasarkan persamaan 2.10

$$c_l^1 = \sum_n \widetilde{h_{2n-l}} c_n^0 + \widetilde{g_{2n-l}} d_n^0 \quad 2.10$$

$$g_n = (-1)^{n+1} \widetilde{h_{-n}}$$

$$\widetilde{g_n} = (-1)^{n+1} h_{-n}$$

Fungsi skala didefinisikan dalam persamaan 2.11 dan 2.12

$$\phi(x) = \sum_n \sqrt{2} \sum_n h_n \phi(2x - n) \quad 2.11$$

$$\widetilde{\phi}(x) = \sqrt{2} \sum_n \widetilde{h_n} \phi(2x - n) \quad 2.12$$

### 2.5.6 Wavelet Transform Haar

*Wavelet Transform Haar* merupakan salah satu algoritma yang dapat digunakan untuk melakukan kompresi sinyal. *Wavelet haar* merupakan *wavelet* yang paling tua dan sederhana [15]. Alasan menggunakan *haar wavelet* adalah memerlukan waktu komputasi yang lebih kecil dari pada transformasi *wavelet* lainnya. Ciri diperoleh dari citra yang telah melewati proses, contoh matriks *haar* ukuran  $L=8 \times 8$ . Baris 1 sampai  $L/2$  *filter low pass* dan baris  $L/2+1$  sampai  $L$  adalah *filter high pass*. Matriks *wavelet haar* ditunjukkan pada Gambar 2.8

>> MatriksHaar(8)							
0.7071	0.7071	0	0	0	0	0	0
0	0	0.7071	0.7071	0	0	0	0
0	0	0	0	0.7071	0.7071	0	0
0	0	0	0	0	0	0.7071	0.7071
0.7071	-0.7071	0	0	0	0	0	0
0	0	0.7071	-0.7071	0	0	0	0
0	0	0	0	0.7071	-0.7071	0	0
0	0	0	0	0	0	0.7071	-0.7071

Gambar 2.9 Matriks Haar ukuran  $8 \times 8$

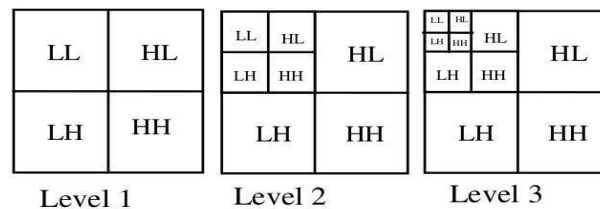
Basis *wavelet* merupakan model yang berbeda-beda dari keluarga *wavelet* yang memiliki kualitas beragam sesuai dengan kriteria yang dimiliki [16]. Pada *wavelet haar*, *wavelet function* didefinisikan dalam persamaan 2.13

$$\psi(t) = \begin{cases} 1 & 0 < t < 0.5 \\ -1 & 0.5 < t < 1 \\ 0 & \text{selainnya} \end{cases} \quad 2.13$$

Untuk *scaling function* pada *wavelet haar* didefinisikan dalam persamaan 2.14

$$\varphi(t) = \begin{cases} 1 & 0 < t < 1 \\ 0 & \text{selainnya} \end{cases} \quad 2.14$$

Dalam proses dekomposisi, *wavelet haar* menerapkan *bank filter* dengan  $h(0) = h(1) = 1/\sqrt{2}$  sebagai koefisien *low-pass* yang menghasilkan citra pendekatan dan  $g_0 = 1/\sqrt{2}$ ,  $g_1 = -1/\sqrt{2}$  sebagai koefisien *high-pass* yang menghasilkan citra detail. Citra hasil dekomposisi akan berukuran seperempat dari ukuran citra sebenarnya. Ilustrasi proses dekomposisi citra dapat dilihat pada Gambar 2.9



Gambar 2.10 Ilustrasi Proses Dekomposisi Citra

Untuk menghitung transformasi *haar wavelet* dari berbagai sampel  $n$  : [17]

1. Menghitung rata-rata setiap pasangan sampel. ( $n / 2$  average).
2. Menghitung perbedaan antara masing-masing rata-rata dan sampel itu dihitung dari ( $n/2$  differences).
3. Tuliskan bagian pertama dari array dengan rata-rata (*averages*).
4. Menulis paruh kedua array dengan perbedaan (*differences*).
5. Ulangi proses pada paruh pertama dari array. Sementara melakukan ini ukuran array harus dibagi dua.

## 2.6 K- Means Clustering

Algoritma *k-means* merupakan salah satu algoritma dengan *partitional*, karena *k-means* didasarkan pada penentuan jumlah awal kelompok dengan mendefinisikan nilai *centroid* awalnya [18]. Algoritma *k-means* menggunakan proses secara berulang-ulang untuk mendapatkan basis data *cluster*. Dibutuhkan jumlah *cluster* awal yang diinginkan sebagai masukan dan menghasilkan titik *centroid* akhir sebagai output. Metode *k-means* akan memilih pola  $k$  sebagai titik awal *centroid* secara acak atau random. Jumlah iterasi untuk mencapai *cluster centroid* akan dipengaruhi oleh calon *cluster centroid* awal secara random. Sehingga didapat cara dalam pengembangan algoritma dengan menentukan *centroid cluster* yang dilihat dari kepadatan data awal yang tinggi agar mendapatkan kinerja yang lebih tinggi [18]

Dalam penyelesaiannya, algoritma *k-means* akan menghasilkan titik centroid yang dijadikan tujuan dari algoritma *k-means*. Setelah iterasi *k-means* berhenti , setiap objek dalam dataset menjadi anggota dari suatu *cluster*. Nilai *cluster* ditentukan dengan mencari seluruh objek untuk menemukan *cluster* dengan jarak terdekat ke objek. Algoritma *k-means* akan mengelompokan item data dalam

suatu dataset ke suatu *cluster* berdasarkan jarak terdekat [19]. Nilai *centroid* awal yang dipilih secara acak yang menjadi titik pusat awal, akan dihitung jarak dengan semua data menggunakan rumus *euclidean distance*. Data yang memiliki jarak pendek terhadap *centroid* akan membuat sebuah *cluster*. Proses ini berkelanjutan sampai tidak terjadi perubahan pada setiap kelompok [18].

Berikut ini langkah-langkah yang terdapat pada *algoritma k-means* [18]:

1. Tentukan  $k$  sebagai jumlah *cluster* yang dibentuk, Untuk menentukan banyaknya *cluster*  $k$  dilakukan dengan beberapa pertimbangan seperti pertimbangan teoritis dan konseptual yang mungkin diusulkan untuk menentukan berapa banyak *cluster*.
2. Tentukan  $k$  *centroid* (titik pusat *cluster*) awal secara random atau berdasarkan keunikan jenis data. Penentuan *centroid* awal dilakukan secara random/acak dari objek objek yang tersedia sebanyak  $k$  *cluster*, kemudian untuk menghitung *centroid cluster* ke- $i$  berikutnya, digunakan rumus seperti pada Persamaan 2.15 dibawah ini.

$$v = \frac{\sum_{i=1}^n X_i}{n}; i = 1, 2, 3, \dots, n \quad 2.15$$

Dimana;

$v$  : *centroid* pada *cluster*

$x_i$  : objek ke- $i$

$n$  : banyaknya objek/jumlah objek yang menjadi anggota *cluster*

3. Hitung jarak setiap objek ke masing-masing *centroid* dari masing-masing *cluster*. Untuk menghitung jarak antara objek dengan *centroid* dapat menggunakan *euclidian distance* seperti pada Persamaan 2.16 dibawah ini.

$$d(x, y) = |x - y| = \sqrt{\sum_{i=1}^n (X_i - Y_i)^2}; i = 1, 2, 3, \dots, n \quad 2.16$$

Dimana;

$x_i$  : objek  $x$  ke- $i$

$y_i$  : daya  $y$  ke- $i$

$n$  : banyaknya objek



4. Alokasikan masing-masing objek ke dalam *centroid* yang paling dekat. Untuk melakukan pengalokasian objek kedalam masing-masing *cluster* pada saat iterasi secara umum dapat dilakukan dengan cara *hard k-means* dimana secara tegas setiap objek dinyatakan sebagai anggota
5. Lakukan iterasi, kemudian tentukan posisi *centroid* baru dengan menggunakan persamaan (2.17)
6. Ulangi langkah 3 jika posisi *centroid* baru jika tidak sama

